Improving personalized ranking in recommender systems with topic hierarchies and implicit feedback

http://www.producao.usp.br/handle/BDPI/48608

Downloaded from: Biblioteca Digital da Produção Intelectual - BDPI, Universidade de São Paulo
Improving Personalized Ranking in Recommender Systems with Topic Hierarchies and Implicit Feedback

Marcelo G. Manzato∗, Marcos A. Domingues∗, Ricardo M. Marcacini†, Solange O. Rezende∗

∗Institute of Mathematics and Computer Science
University of São Paulo
São Carlos, SP, Brazil
{mmanzato, mad, solange}@icmc.usp.br

†Federal University of Mato Grosso do Sul
Três Lagoas, MS, Brazil
ricardo.marcacini@ufms.br

Abstract—The knowledge of semantic information about the content and user’s preferences is an important issue to improve recommender systems. However, the extraction of such meaningful metadata needs an intense and time-consuming human effort, which is impractical specially with large databases. In this paper, we mitigate this problem by proposing a recommendation model based on latent factors and implicit feedback which uses an unsupervised topic hierarchy constructor algorithm to organize and collect metadata at different granularities from unstructured textual content. We provide an empirical evaluation using a dataset of web pages written in Portuguese language, and the results show that personalized ranking with better quality can be generated using the extracted topics at medium granularity.

I. INTRODUCTION

Nowadays, most web sites offer a large number of items (e.g., movies, music, web pages, etc.) to their users, who have to deal with this information overload problem. Recommender systems, which have been emerged in response to this problem, is an information filtering technology used to predict preference ratings of items, not currently rated by the user. In addition, it can be used to output a personalized ranking of items/recommendations that are likely to be of interest to the user [1]. These systems have flourished on the Internet, and web sites such as Amazon1, Netflix2 and Last.fm3 are good examples of recommenders that adapt recommendations to particular user’s tastes.

One of the most popular and successful techniques for recommender systems is Collaborative Filtering (CF) [2]. The main idea behind CF is that users with similar past interests will also share common interests in future. Usually, the CF technique focuses on neighborhood models and latent factors. In the first case, clusters of items are built to recommend items which are similar to the ones preferred by the user in the past. Alternatively, clusters of users can be built to recommend items to a specific user, i.e., items appreciated by other users of similar preferences. In the second case, the recommendation can be computed by uncovering latent associations among users or items by means of factorization techniques [3]. Thereby, an alternative path is comprised to transform both items and users into the same latent factor space, allowing them to be directly comparable [4].

A promising way to improve the performance of recommender systems is to incorporate additional information, such as items’ metadata, besides the typical information about users and items. In this context, we proposed in a previous work [5] the gSVD++ recommender model, which is an extension to the famous SVD++ algorithm proposed by Koren [4]. The gSVD++ consists of incorporating structured item attributes into the factorization process in order to improve the system’s accuracy. However, there are some limitations which can affect its use in practical scenarios:

- Although the model is able to analyze implicit feedback, it still requires explicit information from users (e.g. ratings) to learn its parameters. This is a drawback because explicit feedback is not always available due to cold start, or simply because for some reason, users may not provide any ratings for their preferences.
- When incorporated items’ metadata, a side effect is the need of an intense and time-consuming human effort to identify, collect and label this additional information about the items in order to be properly employed in recommender systems. Moreover, manually label the content become impracticable for large databases.

Thus, for the first limitation, a feasible solution is to rely only on implicit feedback, and adopt some heuristics to infer the user’s preferences indirectly through observing his behavior [6], [7]. For the second limitation, in turn, one possible solution is to use unsupervised learning methods. In this sense, topic hierarchies are efficient models to capture the semantic of textual data in order to organize them [8].

1http://www.amazon.com
2http://www.netflix.com
3http://www.last.fm

1051-4651/14 $31.00 © 2014 IEEE DOI 10.1109/ICPR.2014.635 3696
These models allow the organization of items into topics and subtopics, providing an intuitive way to explore semantic information at different levels of granularity. Topic hierarchies can also be viewed as metadata that characterize the items, and used to better characterize the user’s preferences with respect to the items.

In this paper, we propose a new version of the gSVD++ algorithm, which uses automatically extracted topic hierarchies as metadata. The algorithm is combined with the Bayesian Personalized Ranking technique (BPR) [7], in order to provide a better personalized ranking of items to the users using only implicit feedback. In this way, the main advantage of our technique is that it can work in more realistic scenarios, where additional information from items (structured metadata) and users (explicit feedback) are not always available.

This paper is structured as follows: in Section II we depict the related work; Section III provides a description of gSVD++ [5], BPR [7] and BC² [8] techniques, which are integrated in this proposal. In Section IV we describe our proposal in details; Section V presents the empirical evaluation; and finally, in Section VI we present the final remarks and future work.

II. RELATED WORK

The model proposed in this paper exploits two features in order to generate recommendations with better quality. The first one is the extraction of topic hierarchies, that captures the semantic information from web content, to be used as metadata to characterize items; and the second is the incorporation of such metadata to optimize the personalized ranking of items to the user using only implicit feedback.

Some state-of-the-art approaches for extraction of metadata from web content have been proposed in the literature. In [9], [10], the authors obtain contextual information from online reviews in order to improve item recommendation. For example, in [9] Li et al. compile a list of lexicons and use a string matching method to extract different types of contextual metadata from reviews. In [10], Hariri et al. propose a multi-labeled text classifier based on Labeled Latent Dirichlet Allocation. They assume that there are explicit labels representing contextual information, and such information is obtained for each review by mapping it to the labels. Our proposal exploits an unsupervised method to learn topic hierarchies by analyzing the semantic from web content, and then, the topics are used as metadata to characterize the items. Thus, it does not need a lexicon or a set of labels, normally not available for a web content, to extract the metadata.

In [11], Semeraro et al. propose to use a spreading activation algorithm in order to compute the correlation among terms of a web document and terms from a set of external knowledge sources related to linguistic knowledge, world knowledge, and social knowledge. They use the most correlated external terms as meaningful features/metadata in a content-based recommendation process. An important issue related to this approach is that it can only be used when external knowledge sources are available. Thus, our proposal takes some advantage over this approach since it can be used with internal and external data sources.

Regarding the ranking of items, there is a growing research effort in finding better approaches for optimizing personalized ranking in recommender systems. In [12], Hu et al. propose one of the first methods for personalized ranking in scenarios with implicit feedback. The method extends the matrix factorization by weighting each factorization of user-item interaction with varying of confidence levels. A similar approach that also exploits weighting for the factorizations of user-item is proposed in [13].

The CofiRank algorithm makes use of structured estimation of a ranking loss function based on Normalized Discounted Cumulative Gain (NDCG), and learns the recommendation model by minimizing over a convex upper bound of the loss function [14]. In [15], Shi et al. propose a new context-aware recommendation approach based on Tensor Factorization for Mean Average Precision maximization (TFMAP). To generate top-k recommendations under different types of context, the approach optimizes the metric MAP for learning the model parameters, i.e., latent factors of users, items and context types. In [16], Shi et al. propose the Collaborative Less-is-More Filtering (CLiMF), an approach that learns the model parameters by maximizing the Mean Reciprocal Rank (MRR).

Another approach is the Bayesian Personalized Ranking (BPR) [7], whose optimization criterion is essentially based on pair-wise comparisons between observed and non-observed items. This criterion leads to the optimization of the Area Under ROC Curve (AUC). As we will see in next sections, our proposal uses this framework in order to overcome the main gSVD++ limitation, which is the need of explicit feedback to learn its parameters during training. Our approach considers unstructured metadata of items, i.e., semantic topics automatically extracted from web content, to optimize the personalized ranking of items to the user.

III. BACKGROUND MODELS

Following the same notation in [4], we use special indexing letters to distinguish users, items and metadata: a user is indicated as $u$, an item is referred as $i$, $j$, $k$ and an item’s topic as $g$, $h$. The same notation $r_{ui}$ is used to refer to either explicit or implicit feedback from a user $u$ to an item $i$. In the first case, it is an integer provided by the user indicating how much he liked the content; in the second, it is just a boolean indicating whether the user consumed or visited the content or not. The prediction of the system about the preference of user $u$ to item $i$ is represented by $\hat{r}_{ui}$, which is a floating point score guessed by the recommender algorithm. The set of pairs $(u, i)$ for which $r_{ui}$ is known are represented by the set $K = \{(u, i) | r_{ui} \text{ is known}\}$.

Additional sets used in this paper are: $N(u)$ to indicate the set of items for which user $u$ provided an implicit feedback; $N(u)$ to indicate the set of items that are unknown to user $u$; and $G(i)$ the set of descriptions or topics at certain granularity associated to item $i$.

A. gSVD++

The gSVD++ algorithm [5] is an extension to the original Koren’s SVD++ [4] with the support of items metadata when available. The prediction rule is defined as [5]:

\[ \hat{r}_{ui} = b_{ui} + \left( q_i + \left| G(i) \right| - \alpha \sum_{g \in G(i)} x_g \right) + \left( p_u + \left| N(u) \right| - \beta \sum_{j \in N(u)} y_j \right), \]  
\tag{1}

where each user \( u \) is associated with a user-factors vector \( p_u \in \mathbb{R}^l \), and each item \( i \) with an item-factors vector \( q_i \in \mathbb{R}^l \). The baseline \( b_{ui} \) is defined as \( b_{ui} = \mu + b_u + b_i \) and indicates the difference estimates of users (\( b_u \)) and items (\( b_i \)) in comparison to the overall rating average \( \mu \). The vector of factors \( y_j \in \mathbb{R}^l \) contains the factors for each possible description of item \( i \). A regularization constant \( \alpha \) is set to 1 when there is metadata associated to the item \( i \), and 0 otherwise.

One important disadvantage of gSVD++, as mentioned before, is that it can only analyze implicit feedback only when explicit feedback is available. In addition, the parameter \( x_g \) requires a set of structured descriptions or metadata, which has to be available for each item prone to be recommended. In this way, we propose in this paper a new version of the gSVD++ algorithm in which both limitations are faced by a couple of extensions: in the first case, the model is incorporated to the BPR technique [7], so that recommendation can be computed based only on implicit feedback. In the second case, the unsupervised learning technique BC² [8] is used to extract topic hierarchies directly from the content, and used as metadata by the gSVD++ model. Before we describe the proposed model, we review both BPR and BC² techniques in the next subsections.

### B. Personalized Ranking

Personalized ranking aims at recommending items to the user in decreasing order of his preferences. In many situations, however, these preferences are only provided by implicit feedback from users. This means that the system only knows the positive observations; the non-observed user-item pairs can be either an actual negative feedback or simply the fact that the user does not know about the item’s existence.

In this scenario, Rendle et al. [7] proposed a generic method for learning models for personalized ranking. Instead of training the model using only the user-item pairs, they also consider the relative order between a pair of items, according to the user’s preferences. It is inferred that if an item \( i \) has been viewed by user \( u \) and \( j \) has not (\( i \in N(u) \) and \( j \notin N(u) \)), then \( i >_u j \), which means that he prefers \( i \) over \( j \). Figure 1 presents an example of this method.

To estimate whether a user prefers an item over another, Rendle et al. proposed a technique denominated BPR (Bayesian Personalized Ranking) which is based on a Bayesian analysis using the likelihood function for \( p(i >_u j | \Theta) \) and the prior probability for the model parameter \( p(\Theta) \). The final optimization criterion, BPR-Opt, is defined as:

\[
\text{BPR-Opt} := \sum_{(u,i,j) \in D_K} \ln \sigma(\hat{s}_{uij} - \Lambda_\Theta ||\Theta||^2), \tag{2}
\]

where \( \hat{s}_{uij} := \hat{r}_{ui} - \hat{r}_{uj} \) and \( D_K = \{(u,i,j) | i \in N(u) & j \in N(u)\} \). The symbol \( \Theta \) represents the parameters of the model, \( \Lambda_\Theta \) is the corresponding set of regularization constants, and \( \sigma \) is the logistic function, defined as: \( \sigma(x) = 1/(1 + e^{-x}) \).

**Input:** \( D_K \)

**Output:** Learned parameters \( \Theta \)

Initialize \( \Theta \) with random values

for count = 1,...,\#Iter do

\[
\begin{align*}
\hat{s}_{uij} & \leftarrow \hat{r}_{ui} - \hat{r}_{uj} \\
\Theta & \leftarrow \Theta + \gamma \left( \frac{1}{1 + e^{-\hat{s}_{uij}}} - \hat{p}_{uij} \right) \left( \hat{p}_{uij} - \Lambda_\Theta \right)
\end{align*}
\]

end

**Algorithm 1:** Learning through LearnBPR.

For learning the model, the authors also proposed a variation of the stochastic gradient descent technique, denominated LearnBPR, which randomly samples from \( D_K \) to adjust \( \Theta \). Algorithm 1 shows an overview of the algorithm, where \( \gamma \) is the learning rate.

### C. Topic Hierarchies

In many situations, the items prone to be recommended have additional semantic information available. However, this information is composed by unstructured textual data, making it difficult to be applied to the recommender algorithms. To mitigate this limitation, an interesting strategy is to organize textual information of the items in a topic hierarchy by using unsupervised learning methods. In this case, hierarchical texts clustering algorithms are very useful to automatically organize textual collections in clusters and subclusters – based only on a measure of similarity between textual data. After obtaining the cluster structure, the most important words of each cluster are extracted and used to define topics from texts.

In this work, we use the BC² (Buckshot Consensus Clustering) technique, which was recently proposed by Marcacini et al. [8]. This method is used for unsupervised learning of topic hierarchies from unstructured textual information describing the items (for instance, web page content). Consensus clustering combines different clustering solutions from a same
dataset into a single clustering solution with better quality. For instance, if a textual data item is misplaced in some clustering solution, the same textual data item is not necessarily misplaced in other clustering solutions, thereby consensus clustering can yield to better final solutions.

A brief description of the BC² is as follows: initially, several clustering structures are generated by running various clustering algorithms or alternatively repeated runs of the same algorithm with different parameter values. The generated clusters are aggregated by means of a co-association matrix \( M(l,t) = f_{lt} \), where \( f_{lt} \) is the number of times that textual data items \( l \) and \( t \) are in the same cluster and \( c \) is the number of clustering solutions. In fact, the co-association matrix \( M \) represents a new (robust) concept of proximity among items, and the consensus clustering solution is obtained by applying the UPGMA hierarchical clustering algorithm from the matrix \( M \) [8]. Finally, it is used a feature selection measure called \( F^1 \) to identify the most important words of each (sub) cluster, thereby extracting a topic hierarchy from textual information about the items.

### IV. PROPOSED ALGORITHM

This paper proposes the BPR gSVD++ algorithm, which is a new version of the gSVD++ described in Subsection III-A. In this proposal, we address two limitations of the model, previously discussed: the lack of structured metadata describing every item, and the lack of explicit feedback provided by the users.

In the first case, we inputted the textual content into the topics generator BC² in order to generate a set of topics at a given granularity. These topics, in turn, were used as metadata by the gSVD++ extension. For the topic hierarchy construction, we used different runs of the well-known \( k \)-means algorithm (with random centers initializations and cosine similarity) to obtain several data partitions for the consensus clustering.

In the second case, the gSVD++ was incorporated into BPR in order to optimize the ranking of items according to the user’s implicit preferences. In this manner, the model learns its parameter by measuring the differences between two pairs of items – one known and another unknown –, and the prediction rule \( \hat{r}_{ui} \) defined in Equation 1 now only computes a score indicating how much item \( i \) is relevant to user \( u \).

Considering the prediction rule \( \hat{r}_{ui} \) (Equation 1), we set \( \hat{s}_{uij} = \hat{r}_{ui} - \hat{r}_{uj} \), and define the involved parameters of the model: \( \Theta = \{b_u, p_x, q_y, y_i, x_s\} \). Computing the partial derivatives, we have:

\[
\frac{\partial}{\partial \Theta} \hat{s}_{uij} = \begin{cases} 
1 & \text{if } \Theta = b_i, \\
-1 & \text{if } \Theta = b_j, \\
q_y + \sum_{y \in G(i)} x_y - q_j & \text{if } \Theta = p_u \text{ or } y_j, \\
-\sum_{x \in G(i)} x_h & \text{if } \Theta = q_i \text{ or } x_g, \\
p_u + \sum_{k \in G(j)} y_k & \text{if } \Theta = q_i, \text{ or } x_y, \\
-p_u - \sum_{k \in G(j)} y_k & \text{if } \Theta = x_h, \\
0 & \text{otherwise.}
\end{cases}
\]

It is worth mentioning that \( x_y \) and \( x_h \) are different because the first will iterate through the descriptions of item \( i \) and \( x_h \) with respect to \( j \). Those partial derivatives are used to adjust the parameters of the model, using the BPR technique as described in Subsection III-B. In addition, for each parameter \( \Theta \), a regularization constant \( \lambda_{\Theta} = \{\lambda_b, \lambda_p, \lambda_q, \lambda_y, \lambda_x, \lambda_\Theta\} \) is defined differently using cross-validation. Because it depends on the dataset, the adopted values are presented in next section.

Analyzing Equation 1 again, we note that it uses the parameter \( y_j \), which characterizes the user’s preferences by enhancing the users factors \( p_u \) with the importance of items visited in the past. In addition, the parameter \( x_g \) is used to enhance the items factors \( q_i \) with weights associated to available descriptions. These descriptions, in this paper, refer to the set of topics at given granularity automatically generated by the BC² technique. As we will see in the evaluation, the adoption of both additional parameters is able to improve the results of recommendation.

### V. EMPIRICAL EVALUATION

The empirical evaluation consists of comparing the proposed BPR gSVD++ with two other baselines available in the literature: BPR MF [7] and BPR MF Mapping [17]. The first baseline consists of applying the optimization technique described in Subsection III-B with a traditional factorization model (MF) [3], i.e.:

\[
\hat{r}_{ui} = b_u + p_u^T q_i.
\]

The second baseline, in turn, consists of a method that maps users and items attributes into a metadata awareness matrix factorization model, which is used to reduce cold-start, a well-known problem in recommender systems that happens when new users and items with minimal information cannot be successfully used in the recommendation process [18].

All methods were implemented in the MyMediaLite library [19], and the performance was measured using the following metrics: AUC (Area Under ROC Curve), MAP (Mean Average Precision), Precision at top-5 recommendations and Precision at top-10 recommendations [1]. It is important to mention that AUC does not consider the position of relevant items among the recommendation list [20], differently from MAP. Precision@5 and Precision@10, where the first evaluates the whole recommendation list giving higher scores to top relevant items, and the second and third evaluate only the first 5 and 10 recommended items, respectively.

The experiments were executed with data from a Brazilian website about agrobusiness. It contains 4,659 users and 1,543 different web pages written in Portuguese language. As implicit information, the users generated a total of 15,037 accesses to these pages.

The web pages were used directly to obtain the set of topics. To analyze the effect of the number of topics used as metadata in the recommendation task, we selected subsets of topics using seven different granularities: \{50, 100\}, \{15, 20\}, \{10, 15\}, \{10, 50\}, \{5, 10\}, \{5, 100\} and \{2, 7\}. In the granularity configuration \{m, n\}, the parameter \( m \) identifies the minimum number of items allowed in the topic, while the
parameter $n$ identifies the maximum number of items per topic. When a topic has a few items associated, usually the topic represents more specific semantic information. On the other hand, topics with many items associated represent more general semantic information about the items. Thus, the seven configurations, presented above, generate subsets of 26, 44, 101, 210, 305, 510 and 1290 topics for the considered dataset.

We adopted the 10-fold cross-validation, and compared the proposed algorithm with the baselines using the two-sided paired t-test with a 95% confidence level. The involved constants used with the datasets were empirically defined as: $\#Iter = 40$, $\gamma = 0.05$, $\lambda_f = 0$, $\lambda_p = 0.0025$, $\lambda_y = 0.0025$, $\lambda_{xy} = 0.00025$, $\lambda_y = 20$ and $\lambda_y = 2$.

Figure 2 (a-d) presents the overall results at a varying number of factors using 44 topics extracted from the content. Figure 2 (e-h), in turn, presents the results at different granularities using the best number of factors found for each technique and measure. Such results are also summarized in Table I together with the standard deviation.

From the presented results, we first note that BPR MF obtained the same results regardless of the number of topics, as it does not use any additional information from items. In addition, it is possible to see that the proposed BPR gSVD++ algorithm was able to outperform both baselines at any granularity and number of factors. Moreover, we argue that the best quality recommendation was achieved when 4-11 topics were used to describe the whole content (Figure 2 (e-h)). These sets of topics correspond to a particular level of the hierarchical topic tree, which is not extremely detailed neither generic. It is possible to see that if increased the granularity of the descriptions (i.e., using a higher number of topics to detail the content), it does not improve the recommendation quality. Similarly, if used higher level descriptions (fewer item topics), the results are not improved either. This means that very generic and, at the same time, very specific descriptions cannot improve the set of filtering algorithms presented here; consequently, these descriptions must include a certain level of detailment to achieve the semantics of the content, but not too restricted so that a considerable amount of items cannot be described by a particular topic.

Comparing BPR MF Mapping and BPR gSVD++, it is possible to note that the granularity of descriptions is better explored by our proposal. The BPR MF Mapping also obtained better results when 44 topics were used (green bar in Figure 2 (e) and (h)); however, the BPR gSVD++ was able to achieve more significant improvements on the recommendation quality, in particular when considered the order of relevant items in the recommendation list (blue bar in Figure 2 (f-h)). This is due to the additional parameters $x_y$ and $y_z$: the first parameter has the objective to enhance the item’s factors with relative weights about its descriptions (topics); the second, on the other hand, characterizes the user’s factors with relative weights about the set of items he visited in the past (implicit feedback).

In summary, in this evaluation we showed how much the performance of recommendation strategies can be improved if metadata awareness and implicit feedback are incorporated to compute personalized ranking. We also demonstrated at which level of detail or granularity these descriptions must be in order to provide better quality recommendation using the set of models presented in this work.

VI. CONCLUSION AND FUTURE WORK
This paper proposed a new version of the gSVD++ algorithm [5] which addresses two limitations of the model: the lack of structured metadata describing every item, and the lack of explicit feedback provided by the users. In the first case, we use an unsupervised topic hierarchy constructor technique to organize and collect metadata at different granularities from unstructured textual content. In the second case, we incorporated the model into the BPR technique [7] so that only implicit feedback is necessary to gather the preferences of users.

We performed an empirical evaluation to demonstrate how much the topic hierarchy constructor can improve the recommendation quality. The experiments were executed with a dataset in Portuguese language from agrobusiness domain, and the results showed that, for the considered models, the granularity of descriptions cannot be too generic, and either, too specific. In the first case, generic descriptions capture the general idea of the content, but inside a given category, subtopics can be extracted, which may be differently preferred by the user. On the other hand, when too specific descriptions are used, the recommendation algorithms are prone to over-specialization, in which case a topic appreciated by the user is unable to retrieve other items which are semantically related.

As future work, we plan to improve the BPR algorithm by exploring the hierarchy of topics generated by BC² technique. Such study will also include the incorporation of biases towards the main categories of the content. The idea is to infer relative preferences of two items in case both are known or unknown by the user.

ACKNOWLEDGMENT
The authors would like to thank the financial support from CAPES, CNPq and FAPESP (grant #2013/25547-1 and #2012/13830-9).

REFERENCES
Fig. 2. Comparison of the models. Graphs (a-d) present, respectively, the AUC, MAP, Precision@5 and Precision@10 of the considered techniques at a varying number of factors using 44 topics ([15, 20]). Graphs (e-h) show the results for the same measures at different granularities using the best number of factors for each technique and measure.

TABLE I. COMPARISON OF THE MODELS. VALUES FOR BPR gSVD++ COMPARED TO BASELINES ARE STATISTICALLY SIGNIFICANT (P-VALUE < 0.05)

<table>
<thead>
<tr>
<th># topics</th>
<th>BPR MF</th>
<th>BPR MF Mapping</th>
<th>BPR gSVD++</th>
<th>BPR MF</th>
<th>BPR MF Mapping</th>
<th>BPR gSVD++</th>
<th>BPR MF</th>
<th>BPR MF Mapping</th>
<th>BPR gSVD++</th>
</tr>
</thead>
<tbody>
<tr>
<td>26</td>
<td>0.9585 ± 0.001</td>
<td>0.9586 ± 0.001</td>
<td>0.9717 ± 0.001</td>
<td>0.1313 ± 0.004</td>
<td>0.1306 ± 0.003</td>
<td>0.1684 ± 0.004</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>0.9585 ± 0.001</td>
<td>0.9608 ± 0.0006</td>
<td>0.9749 ± 0.0008</td>
<td>0.1313 ± 0.004</td>
<td>0.1327 ± 0.002</td>
<td>0.1912 ± 0.007</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>101</td>
<td>0.9585 ± 0.001</td>
<td>0.9595 ± 0.0007</td>
<td>0.9728 ± 0.0009</td>
<td>0.1313 ± 0.004</td>
<td>0.1307 ± 0.002</td>
<td>0.1768 ± 0.005</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>210</td>
<td>0.9585 ± 0.001</td>
<td>0.9585 ± 0.001</td>
<td>0.9712 ± 0.001</td>
<td>0.1313 ± 0.004</td>
<td>0.1299 ± 0.002</td>
<td>0.1651 ± 0.005</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>305</td>
<td>0.9585 ± 0.001</td>
<td>0.9586 ± 0.0009</td>
<td>0.9717 ± 0.001</td>
<td>0.1313 ± 0.004</td>
<td>0.1323 ± 0.005</td>
<td>0.1729 ± 0.004</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>510</td>
<td>0.9585 ± 0.001</td>
<td>0.9586 ± 0.001</td>
<td>0.9709 ± 0.0009</td>
<td>0.1313 ± 0.004</td>
<td>0.1297 ± 0.002</td>
<td>0.1670 ± 0.007</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1230</td>
<td>0.9585 ± 0.001</td>
<td>0.9586 ± 0.001</td>
<td>0.9715 ± 0.001</td>
<td>0.1313 ± 0.004</td>
<td>0.1302 ± 0.002</td>
<td>0.1689 ± 0.006</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>


